

A NOVEL COVID-19 CLASSIFICATION METHOD BASED ON CURE CLUSTERING

Bergen KARABULUT^{1*}, Güvenç ARSLAN², Halil Murat ÜNVER³

¹Informatics and Information Security Research Center (BİLGEM), TUBITAK, Türkiye

²Department of Statistics, Kırıkkale University, Türkiye

³Department of Computer Engineering, Kırıkkale University, Türkiye

ABSTRACT: COVID-19 is a serious disease that spreads rapidly and affects the world. Alternative methods based on machine learning are recommended to diagnose COVID-19 positive and negative cases cheaper and faster. However, as the data size increases, problems such as space requirement or classification time may arise. KNN (K-nearest neighbor), a simple but effective machine learning method, is widely used in various fields. However, the effectiveness of the KNN algorithm decreases considerably when the sample size is large and the number of features is too large. To solve these problems, it is important to use datasets more effectively and to select meaningful parts of the data. The current study proposes an improved neighborhood-based classification method called CURE-NN and compares its performance with standard NN and KNN algorithms. The proposed CURE-NN method obtains reduced structural information from the data by applying clustering before classification to use the dataset more effectively. The resulting reduced structural information was used as a training set in the classification process. The proposed method was applied to the COVID-19 dataset. With this method, while the classification success is preserved as much as possible compared to the NN and KNN methods, the data used in the test phase is reduced by up to 96%. Experimental results show that the reduced data obtained based on structural information can be used instead of the entire data set. In addition, the method works by using only one neighbor, thus eliminating the need for the K parameter compared to the KNN algorithm.

Keywords: Coronavirus, Covid-19 Diagnosis, K-Nearest Neighbor, Cure Clustering, Classification.

1. INTRODUCTION

In December 2019, the city of Wuhan, Hubei Province of China, became the center of an epidemic of pneumonia of unknown origin. By January 7, 2020, Chinese scientists had isolated a novel coronavirus (CoV) from patients in Wuhan [1]. This new disease has been named coronavirus disease (COVID-19) by WHO, and the virus that causes this disease is identified as severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) [2]. Over ten thousand people were infected and hundreds died within a month [3]. This epidemic spread very quickly around the world, and on March 11, 2020, WHO officially declared the epidemic caused by COVID-19 as a Pandemic [4]. As of August 2, 2023, there appear to be 768,983,095 confirmed cases of COVID-19, including 6,953,743 deaths reported to WHO globally [5]. SARS-CoV-2 is transmitted from person to person through close contact and causes COVID-19 [6]. Therefore, early diagnosis of the disease is crucial not only for individual patient care related to rapid administration of treatment but also for adequate patient isolation from a broader public health perspective. Laboratory confirmations of SARS-CoV-2 were performed with a virus-specific reverse transcriptase-polymerase chain reaction (RT-PCR) test, but this test can take up to 2 days to complete [7]. In many places, detection of COVID-19 is conducted by RT-PCR tests [8]. Despite its known shortcomings, real-time reverse transcriptase-polymerase chain reaction (rRT-PCR) is the current gold standard for confirming infection. Some of these shortcomings are; long turnaround times (results are produced in 3-4 hours), lack of potential reagents, high false-negative rates of 15-20%, and the need for certified laboratories, expensive equipment, and trained personnel [9]. Therefore, there is a need for faster, cheaper, and more accessible alternative methods. Accordingly, researchers have used Chest CT scans [7, 10, 11] or Chest X-Ray images [8, 12, 13] on the diagnosis of COVID-19. As it is known, machine learning methods are widely used on health data [14-18]. However, some studies apply machine learning approaches for COVID-19, which has attracted a lot of attention recently. For example, Arpacı et al. [19] developed 6 predictive models based on 14 clinical features using 6 different classifiers, namely BayesNet, Logistic, IBk, CR, PART, and J48, for the diagnosis of COVID-19. In another study, Khakharia et al. [3] developed a COVID-19 outbreak prediction system for the top 10 high and densely populated countries. On the other hand, with the gradual growth of medical and health care data, classification methods for traditional medical health big data have problems such as large sample sizes and slow processing [20].

In the literature, various studies have been carried out using the machine learning method to identify COVID-19 positive cases. Brinati et al. [9] developed two machine learning classification models to distinguish between SARS-CoV-2 positive or negative patients using hematochemical values (i.e., white blood cell counts and platelets, CRP, AST, ALT, GGT, ALP, LDH plasma levels) from routine blood examinations. Ahamad et al. [21] developed a machine learning

*Corresponding Author. Email: brgnkarabulut@gmail.com

Received Date: 29/03/2024

Accepted Date: 26/06/2024

This work is licensed under a Creative Commons Attribution 4.0 License.

For more information, see <https://creativecommons.org/licenses/by-sa/4.0>



methodology to identify the most important and most notable clinical symptoms predicting true COVID-19 positive cases. Hamed et al. [22] proposed a new variant of KNN, called KNNV, to classify COVID-19 in IHC datasets. The researchers used rough set-theoretic techniques to address both incompleteness and heterogeneity, as well as to find an ideal value for K. Sun et al. [23] developed an XGBoost-based classification model by integrating multi-omics data to examine subtle changes in gene expression and pathways of COVID-19 patients with different severity levels. The researchers suggested that they were able to clearly distinguish patients from different severity groups and accurately predict the pathological condition by reaching a high micro-average AUROC and micro-average AUPR of 0.9941 and 0.9837, respectively. Arpacı et al. [19] developed 6 predictive models based on 14 clinical features using 6 different classifiers, namely BayesNet, Logistic, IBk, CR, PART, and J48, for the diagnosis of COVID-19. Zoabi et al. [24] proposed a machine learning model that predicts SARS-CoV-2 infection positive in an RT-PCR test by asking eight key questions. Viana dos Santos Santana et al. [25] aimed to effectively prioritize symptomatic patients in the testing process to aid in the early detection of COVID-19 in Brazil. Raw data from 55,676 Brazilians were preprocessed and chi-square testing was conducted to verify the suitability of gender, health professional, fever, sore throat, dyspnea, olfactory disorders, cough, coryza, taste disorders, and headache characteristics. After preprocessing, multilayer perceptron, gradient boosting machine, decision tree, random forest, extreme gradient boosting, K-nearest neighbors, support vector machine, and logistic regression classification algorithms were applied.

Although the K-nearest neighbor (KNN) method, one of the well-known machine learning classification methods, is simple, it has proven to be quite efficient and effective for solving various classification problems in real life [26]. However, space requirement and classification time problems come to the fore in nearest neighbor-based classifiers. That is, in these classifiers, it is necessary to store the entire training set and search to classify a particular sample [27]. For the solution to such problems, approaches that detect structural information from the data set based on pre-processes such as clustering to benefit from the data more effectively [28-31] draw attention. There is also a lot of data for COVID-19, and it continues to increase. Therefore, there is a need for approaches that will enable more effective use of huge COVID-19 data.

In line with the mentioned developments, in this study, a new classification approach that makes use of the structural information of the data set has been investigated to benefit more effectively from the increasing data. CURE clustering method, a successful clustering method, was used to extract structural information from the dataset, in other words, to reduce data. A classification approach has been developed that uses the reduced samples obtained by the clustering process instead of the training set and works with the nearest neighbor approach. The developed approach has been applied to COVID-19 data. The main contributions of this study are: First, the contribution of the structural information provided by the clustering process to the classification problem has been investigated to use the increasing data more effectively. In this line, an advanced neighborhood algorithm including pre-classification clustering has been proposed. The proposed approach was applied to a real dataset of the COVID-19 pandemic, which affected the whole world, and was compared with standard NN and KNN approaches. Secondly, with the clustering process applied in the proposed approach, the most meaningful and reduced structural information is obtained from the data set and this information is used in the classification phase instead of the entire training set. In this way, less data is stored and fewer searches are made for the testing phase compared to standard nearest neighbor approaches, contributing to the speed of testing phase. In addition, the proposed approach uses one neighbor in the label assignment of test samples, thereby eliminating the K parameter involved in the K-nearest neighbors method. While increasing the efficiency of the standard KNN approach, classification accuracy can be maintained as much as possible.

2. MATERIALS AND METHODS

2.1. Datasets

The original COVID-19 dataset provided by Viana dos Santos Santana et al. [32] includes early-stage symptoms, comorbidities, demographics, and descriptions of symptoms of patients tested. Patients were tested using viral or rapid testing. Raw data were collected from the public health agency of the city of Campina Grande, State of Paraíba in Northeast Brazil. The researchers preprocessed this dataset; selecting only completed tests, marking them as positive or negative, applying string matching algorithms to correct some inconsistencies, and removing rows from recurrent and asymptomatic patients. They also focused on the most frequent and relevant demographics and reported early-stage symptoms to select features. Using the NearMiss algorithm, they balanced the data by taking into account positive and negative cases by performing random undersampling. The features included in this dataset are 'Symptom-Throat Pain',

‘Symptom-Dyspnea’, ‘Symptom-Fever’, ‘Symptom-Cough’, ‘Symptom-Headache’, ‘Symptom-Taste Disorders’, ‘Symptom-Olfactory Disorders’, ‘Symptom-Gender’, and ‘Are you a health professional?’.

In the current study, the 5th version of the related dataset in Mendeley Data was used. This version includes Rapid and PCR datasets and datasets created by combining both. In addition, the unbalanced state of each data set and the balanced state according to the class label can be accessed. There are no missing values in the presented datasets. These datasets, along with the sample numbers and sample distributions by class, are given in Table 1.

Table 1. Balanced and unbalanced COVID-19 datasets

Dataset Name	Instances	Class (#of instances)
rapid_balanced	1296	0 (648), 1 (648)
rapid_unbalanced	17242	0 (648), 1 (16594)
pcr_balanced	1832	0 (916), 1 (916)
pcr_unbalanced	2779	0 (916), 1 (1863)

As seen in Table 1, unbalanced datasets are imbalanced towards negative cases (1). In the experimental analysis part of this study, tests were conducted on both balanced and unbalanced versions of these datasets.

2.2. Evaluation Criteria

Accuracy, Precision, Recall, and F1-Score, which are commonly used evaluation criteria, were used to evaluate the results. In these metrics, True Positive (TP) and True Negative (TN) indicate the number of correctly classified positive and negative samples, and False Positive (FP) and False Negative (FN) indicate the number of incorrectly classified negative and positive samples, respectively. Accuracy provides an overall measure of how accurately the model makes predictions across the entire dataset. The purpose of Precision is to evaluate TP entities in relation to FP entities, whereas the purpose of Recall is to evaluate TP entities in relation to FN entities. The F₁score represents the harmonic mean of precision and recall. The formulations of these metrics are given in Equation (1)-(4) [38]. In addition to these criteria, the number of samples used in the training set by the methods and the test time of each method was evaluated.

$$Accuracy(Acc) = (TP + TN)/(TP + FP + TN + FN) \quad (1)$$

$$Precision = TP/(TP + FP) \quad (2)$$

$$Recall = TP/(TP + FN) \quad (3)$$

$$F_1score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

2.3. CURE Clustering Algorithm

CURE algorithm is a hierarchical agglomerative clustering method introduced by Guha et al. [33]. In this method, a fixed number c is first determined for well-scattered points in a cluster. Scattered points capture the shape and extension of the cluster. The selected scattered points are then shrunk with a fraction α towards the center of the cluster. Scattered points after shrinking are used as representative points. The cluster pairs with the closest representative points are the clusters that are combined in each step of the CURE hierarchical clustering algorithm. For large databases, CURE implements random sampling and partitioning. The main steps of the CURE algorithm are as follows:

Traditional clustering methods either create clusters of spherical shape and similar size or are very sensitive to outliers. The CURE clustering method, on the other hand, is more robust to outliers and identifies clusters of different sizes and non-spherical shapes. CURE can find clusters of arbitrary shapes and sizes because it represents each cluster with multiple representative points. In addition, shrinking the representative points towards the center allows the method to avoid problems associated with noise and outliers [34].

2.4. K-Nearest Neighbor (KNN)

One of the supervised learning methods is Instance-Based Learning, also called Lazy Learning [35]. The classic example of the instance-based learning method is the K-nearest neighbor (KNN) classification algorithm [36]. The nearest neighbor (NN) classifier assigns the class of its nearest neighbor in the training set according to the distance function to a given test pattern. The K-nearest neighbor classifier, where K is an integer and $K \geq 1$, is a generalization of the nearest

neighbor classifier [27], and the KNN algorithm works by classifying each new instance among its K-nearest neighbors by majority label. The naive implementation of the nearest neighbor rule requires storing all previously classified data points and then comparing each stored point to classify each sample point [37].

3. THE PROPOSED CLASSIFICATION METHOD

In this section, a new neighborhood-based classification method called CURE-NN is proposed. This method consists of two steps: data reduction and classification. To present the steps of the proposed CURE-NN method, the sample dataset, whose scatter plot is given in Figure 1a, is used. This dataset contains two classes and only two attributes of the dataset are used to present the results visually. Random test samples were selected from the sample dataset, and the remaining samples were used as the training set (Figure 1b).

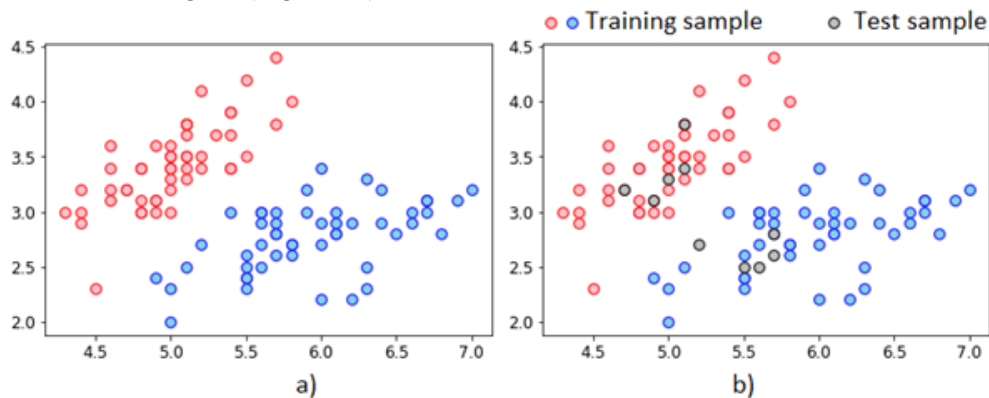


Fig. 1. a) Sample training set b) Training and test samples

The CURE-NN method first applies data reduction. The steps of this process are given below.

- i. The CURE-NN method first applies data reduction. The steps of this process are given below. The training set is divided into subsets based on the available class labels. There are two classes in the dataset. They are divided into subsets as the first class $Subset_1$ and the second class $Subset_2$. These subsets are shown in Figure 2a.
- ii. Class labels are deleted from each resulting subset (Figure 2b). In this way, the subsets are ready for the application of clustering, which allows to extract structural information from unlabeled data.
- iii. CURE clustering is applied to each subset separately. For the CURE clustering process, parameters such as the shrink factor α , the number of clusters k , and the number of representative points c should be determined. In the proposed CURE-NN method, since each class is clustered separately by making use of class information, classes are considered as a single cluster. In this way, this parameter is fixed as $k = 1$ for CURE-NN and does not need to be tuned. The shrink factor α is used as $\alpha = 0.2$ for this example. If the number of representative points c , which is another parameter, is determined by $c = n_{class_i} * DR_{rate}$, with the number of samples in the i .th subset n_{class_i} and data reduction rate DR_{rate} . For the sample dataset, $DR_{rate} = 0.4$ is applied, so in this example, 40% of the sample in each class will be chosen as representative points. In this way, data reduction was made so that 40% of the data in each class remains. Representative points obtained after applying CURE clustering to each subset with the specified parameters are retained (Figure 2c).
- iv. Class labels are added to representative points (Figure 2d).
- v. The representative points are merged to form the new training set (Figure 2e)

In the classification phase of the CURE-NN method, a neighborhood-based classification approach is applied. Instead of the entire training set in the classification stage, the reduced data in the first stage of the CURE-NN method is used as the training set. With reduced training data, the class label of the closest sample is assigned to each test sample. In other words, the class is determined by looking at the neighbor to which each sample is closest. Figure 2f shows the determination of the nearest neighbor for two test samples.

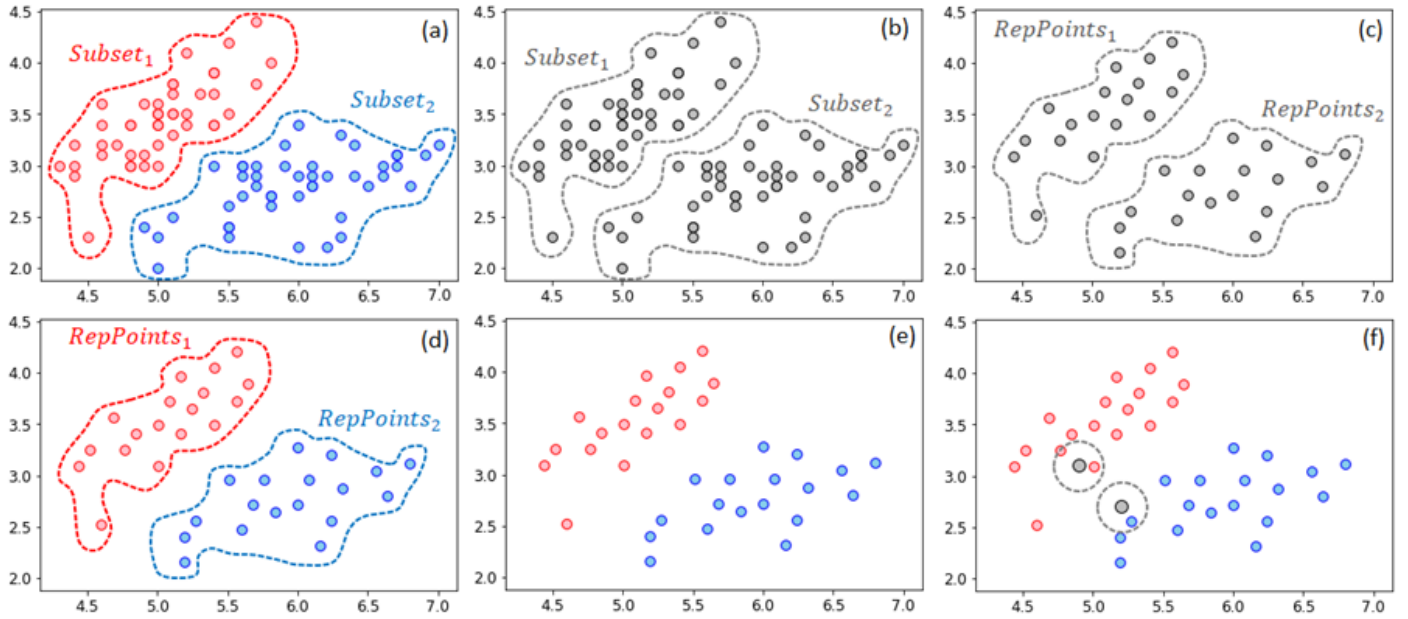


Fig. 2. Visual representation of CURE-NN steps **a)** Training dataset with two classes, **b)** Subsets, **c)** Representative points, **d)** Representative points with class labels, **e)** New training set, **f)** Nearest neighbor for two test samples

The pseudocode of the proposed CURE-NN classification method is as follows:

Pseudocode of the proposed CURE-NN Algorithm

Input:	$X = \{x_i \in \mathbb{R}^d\}_{i=1}^n$, training set with n samples
Output:	y' , the class label of the test sample
Step 1:	Set/determine the shrink factor α and the data reduction rate DR_{rate}
Step 2:	Divide the training set into subsets according to the class labels, with the class labels in the dataset $L = \{l_1, l_2, l_3, \dots, l_t\}$. for all $l_i \in L$ do $X_{Subset_i} = \{x_k \in X: x_k \text{ belongs to class } l_i\}$ end for
Step 3:	Remove class labels from subsets. for $i = 1$ to t do Remove class label l_i from X_{Subset_i} end for
Step 4:	Find representative points by applying CURE to each subset, $len()$: the function that returns the number of samples in the given set for all $i = 1$ to t do $c = len(X_{Subset_i}) * DR_{rate}$ $RepPoints_i = CURE(X_{Subset_i}, c, \alpha)$ end for
Step 5:	Add class labels to representative points for $i = 1$ to t do Add class label l_i to $RepPoints_i$ end for
Step 6:	Merge the representative points; this union creates the new training set. $X^{new} = \bigcup_{i=1}^t RepPoints_i$
Step 7:	Calculate the distance between each instance in x' and X^{new} for a given query example $z = \{x', y'\}$ $m = len(X^{new})$

for all $x_i \in X^{new}$ do
 Calculate $d(x', x_i)$, $i = 1, 2, 3, \dots, m$; where d denotes the Euclidean distance between points.
 end for

Step 7: Select $x_z \in X^{new}$ such that $dist(x', x_z) = \min_j dist(x', x_j)$.

Step 8: Assign class label of x_z to the query x' .
 $y' = y_z$

Here, we note that Karabulut et al. [31] used the CURE clustering method in a similar way. They have developed a method called RP-SVM by combining the CURE clustering algorithm with the SVM method. In the current study, unlike in Karabulut et al., the reduced data from CURE is adapted to a simpler approach, an NN-based approach. In addition, this study used COVID-19 data, which is a larger dataset compared to datasets in Karabulut et al. [31].

4. EXPERIMENTAL RESULTS

In the experimental analysis part of this study, the CURE-NN method was analyzed comparatively with NN and KNN methods. The indicated models were applied on the rapid_balanced, rapid_unbalanced, pcr_balanced, pcr_unbalanced, both_test_balanced, and both_test_unbalanced COVID-19 datasets described in the previous sections. Some parameters need to be adjusted for the applied methods. For the KNN method, only the number of neighbors - the K parameter - should be determined. For parameter K, $\{1, 2, 3, \dots, 30\}$ values are applied. For the CURE-NN method, some parameters need to be determined only for the pre-stage where CURE is applied. These parameters are the number of clusters k , the shrink factor α and the number of representative points c . In the CURE-NN method, since each class is clustered separately by using the class information, the classes are handled as a single cluster. In this approach, $k = 1$ is fixed, eliminating the need to specify the parameter. For the shrink factor α , Guha et al. [33] found that the 0.2-0.7 value range is suitable for defining non-spherical clusters while reducing the effect of outliers. In the same study, $\alpha = 0.3$ was used as the default value for α . The default value of $\alpha = 0.3$ is used for CURE-NN. If the number of representative points c , which is another parameter, is determined by $c = n_{class_i} * DR_{rate}$, with the number of samples in the i .th subset n_{class_i} and data reduction rate DR_{rate} . To reduce the data as much as possible, $DR_{rate} = 0.2$ was applied in the experimental analysis part, that is, 20% of the number of samples in each class of the dataset is the number of points that will represent the relevant class.

Models were trained and tested using stratified nested 10-fold cross-validation. In the standard nested cross-validation (nested CV) approach, the data is split into k outer folds and then inner folds are created in each outer training set for feature selection, parameter setting, and training of models [39]. It has been found that Nested Cross Validation considerably reduces bias [40] and can be used to obtain reliable classification accuracy [39]. The results of the analysis were evaluated with the Accuracy, Precision, Recall, and F-score criteria specified in subsection 3.3. In addition to these criteria, evaluations were made in terms of the number of samples used by the methods in the training set n_{Train} and the test time of each method, t_{Test} .

4.1. Tests on Balanced Datasets

In this case the proposed CURE-NN, NN, and KNN methods have been applied to balanced COVID-19 datasets. In the application, the stratified nested 10-fold cross-validation process was repeated 30 times with different seed values. The obtained results were averaged. The results are given in Table 2.

Table 2. Results from balanced datasets

Dataset	Indices	NN	KNN	CURE-NN
pcr_balanced	Accuracy	0.9569	0.9567	0.9596
	Precision	0.9678	0.9654	0.9643
	Recall	0.9476	0.9495	0.9557
	F-score	0.9574	0.9571	0.9598
	n_{Train}	1649	1649	229.61
	t_{Test}	0.0056	0.0077	0.0046
	Accuracy	0.9312	0.9265	0.9179

rapid_balanced	Precision	0.9668	0.9464	0.9438
	Recall	0.9036	0.9115	0.8992
	F-score	0.9337	0.9280	0.9202
	n_{Train}	1167	1167	234.00
	t_{Test}	0.0064	0.0047	0.0034
	both_test_balanced	Accuracy	0.8695	0.8808
Precision		0.9115	0.9310	0.8808
Recall		0.8415	0.8466	0.8532
F-score		0.8748	0.8865	0.8664
n_{Train}		2816	2816	564
t_{Test}		0.0123	0.0132	0.0096

n_{Train} : number of training set samples, t_{Test} : test time in seconds

As seen in Table 2, for the **pcr_balanced dataset**, each fold contains 1649 training and 183 test samples. While the NN and KNN methods use all training samples, the CURE-NN method uses only about 14% of the training set, with an average of approximately 229.61 samples. The CURE-NN method considerably reduced the training sample in this dataset and nevertheless obtained better accuracy than the NN and KNN methods. In addition, the test time of the CURE-NN method is shorter compared to other methods. For the **rapid_balanced dataset**, each fold contains 1167 training and 129 test samples. While the NN and KNN methods use all training samples, the CURE-NN method uses only about 18% of the training set, with an average of about 234 samples. The CURE-NN method considerably reduced the training sample in this dataset, but still achieved close accuracy to the NN and KNN methods. In addition, the test time of the CURE-NN method is shorter compared to other methods. For the **both_test_balanced dataset**, there are 2816 training and 312 test samples in each fold. While the NN and KNN methods use all training samples, the CURE-NN method uses only about 20% of the training set, with an average of about 564 samples. The CURE-NN method considerably reduced the training sample in this dataset, but still achieved close accuracy to the NN and KNN methods. In addition, the test time of the CURE-NN method is shorter compared to other methods.

4.2. Tests on Unbalanced Datasets

The proposed CURE-NN, NN and KNN methods were applied on COVID-19 datasets given as unbalanced. In the application, nested 10fold cross-validation is applied. The results obtained are given in Table 3.

Table 3. Results from unbalanced datasets

Dataset	Indices	NN	KNN	CURE-NN
pcr_unbalanced	Accuracy	0.9673	0.9655	0.9629
	Precision	0.9443	0.9377	0.9563
	Recall	0.9561	0.9568	0.9334
	F-score	0.9500	0.9469	0.9446
	n_{Train}	2502	2502	264.1
	t_{Test}	0.0107	0.0119	0.0074
rapid_unbalanced	Accuracy	0.9890	0.9887	0.9890
	Precision	0.8395	0.7914	0.8827
	Recall	0.8654	0.9031	0.8363
	F-score	0.8515	0.8403	0.8582
	n_{Train}	15518	15518	590.9
	t_{Test}	0.1335	0.1486	0.0456
both_test_unbalanced	Accuracy	0.8841	0.9305	0.8569
	Precision	0.5587	0.3618	0.6874
	Recall	0.3634	0.6180	0.3115
	F-score	0.4327	0.4482	0.4285
	n_{Train}	18019	18019	726.9
	t_{Test}	0.1664	0.2542	0.0516

n_{Train} : number of training set samples, t_{Test} : test time in seconds

For the **pcr_unbalanced dataset**, there are 2502 training and 277 test samples in each fold. While the NN and KNN methods use all training samples, the CURE-NN method uses only about 11% of the training set, with an average of about 264.1 samples. The CURE-NN method considerably reduced the training sample in this dataset, but still achieved similar accuracy to the NN and KNN methods. In addition, the test time of the CURE-NN method is shorter compared

to other methods. For the **rapid_unbalanced dataset**, each fold contains 15518 training and 1724 test samples. While the NN and KNN methods use all training samples, the CURE-NN method uses only about 4% of the training set, with an average of about 590.9 samples. The CURE-NN method considerably reduced the training sample in this dataset and nevertheless obtained the best accuracy value. In addition, the test time of the CURE-NN method is shorter compared to other methods. For the **both_test_unbalanced dataset**, each fold contains 18019 training and 2002 test samples. While the NN and KNN methods use all training samples, the CURE-NN method uses only about 4% of the training set, with an average of about 726.9 samples. The CURE-NN method considerably reduced the training sample in this dataset but caused a decrease in the accuracy value. In addition, the test time of the CURE-NN method is shorter compared to other methods. Taken together, as seen in Figure 3, the CURE-NN method used only a very small part of the training data in all datasets. In addition, it is seen that the test times are shortened.

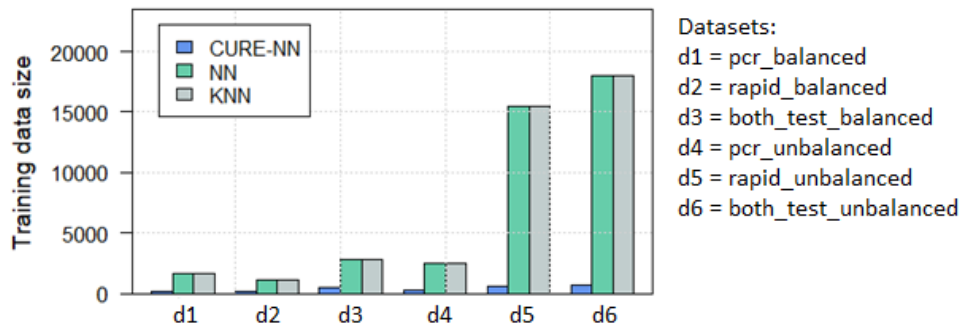


Fig. 3. Comparison of training data size

In terms of classification accuracy, the results are given comparatively in Figure 4 with the box-plot graph. This graph shows no considerable differences between the methods.

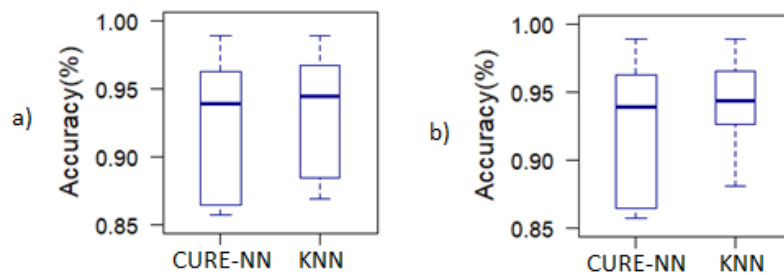


Fig 4. a) Comparison of classification accuracy between the CURE-NN and NN, **b)** Comparison of classification accuracy between the CURE-NN and KNN

The results of the current study show that with the proposed approach, structural information can be extracted from the COVID-19 dataset quite successfully, and effective classification can be made with the reduced data in this way.


In the literature, there are studies aimed at increasing classification effectiveness by integrating clustering methods such as k-means, BIRCH, k-spatial medians, and CURE into classification methods [41-48, 31]. It is generally observed that clustering methods are adapted to the SVM in these studies. In this study, however, the K-nearest neighbor classification method, which is one of the commonly used classification methods, has been addressed. To enhance its classification effectiveness, the effective clustering method CURE has been utilized.


5. CONCLUSION


The size of COVID-19 data has been increasing. As it is known, the increase in data size brings problems such as space requirement or an increase in classification time. In the current study, a classification approach based on reduced structural information has been investigated to effectively represent the dataset to solve these problems. CURE clustering approach is used to obtain reduced structural information from the dataset. Classification is applied by using reduced structural information instead of the entire data set. Classification is done with a neighborhood-based approach. This new method, called CURE-NN, has been applied to balanced and unbalanced COVID-19 data. It has been observed that the CURE-NN method can maintain classification accuracy while reducing data from 80% to 96% and even achieve

better accuracy in some datasets. The results show that with the proposed approach, structural information can be successfully extracted from the dataset, enabling effective classification with reduced data.

6. ORCID

Bergen KARABULUT  <https://orcid.org/0000-0003-0755-1289>

Güvenç ARSLAN  <https://orcid.org/0000-0002-4770-2689>

Halil Murat ÜNVER  <https://orcid.org/0000-0001-9959-8425>

REFERENCES

- [1]. Wang, C., Horby, P.W., Hayden, F. G., & Gao, G.F. (2020). A novel coronavirus outbreak of global health concern. *The lancet* 395(10223), 470-473.
- [2]. World Health Organization. Naming the coronavirus disease (COVID-19) and the virus that causes it from [https://www.who.int/emergencies/diseases/novel-coronavirus-2019/technical-guidance/naming-the-coronavirus-disease-\(covid-2019\)-and-the-virus-that-causes-it](https://www.who.int/emergencies/diseases/novel-coronavirus-2019/technical-guidance/naming-the-coronavirus-disease-(covid-2019)-and-the-virus-that-causes-it), accessed on 2023-08-18.
- [3]. Khakharia, A., Shah, V., Jain, S., Shah, J., Tiwari, A., ... & Mehendale, N. (2021). Outbreak prediction of COVID-19 for dense and populated countries using machine learning. *Annals of Data Science* 8(1), 1-19.
- [4]. WHO Director-General's opening remarks at the media briefing on COVID-19 - 11 March 2020, from <https://www.who.int/director-general/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-covid-19---11-march-2020>, accessed on 2023-08-18.
- [5]. WHO Coronavirus (COVID-19) Dashboard, from <https://covid19.who.int/>, accessed on 2023-08-09.
- [6]. Chu, D.K., Akl, E.A., Duda, S., Solo, K., Yaacoub, S., Schünemann, H.J., ... & Reinap, M. (2020). Physical distancing, face masks, and eye protection to prevent person-to-person transmission of SARS-CoV-2 and COVID-19: a systematic review and meta-analysis. *The lancet* 395(10242), 1973-1987.
- [7]. Mei, X., Lee, H.C., Diao, K.Y., Huang, M., Lin, B., Liu, C., ... & Yang, Y. (2020). Artificial intelligence-enabled rapid diagnosis of patients with COVID-19. *Nature medicine* 26(8), 1224-1228.
- [8]. Madaan, V., Roy, A., Gupta, C., Agrawal, P., Sharma, A., Bologa, C., & Prodan, R. (2021). XCOVNet: Chest X-ray Image Classification for COVID-19 Early Detection Using Convolutional Neural Networks. *New Generation Computing* 1-15.
- [9]. Brinati, D., Campagner, A., Ferrari, D., Locatelli, M., Banfi, G., Cabitza, F. (2020). Detection of COVID-19 infection from routine blood exams with machine learning: a feasibility study. *Journal of medical systems* 44(8), 1-12.
- [10]. Li, L., Qin, L., Xu, Z., Yin, Y., Wang, X., Kong, B., ... & Xia, J. (2020). Using artificial intelligence to detect COVID-19 and community-acquired pneumonia based on pulmonary CT: evaluation of the diagnostic accuracy. *Radiology* 296(2), E65-E71.
- [11]. Wang, S., Kang, B., Ma, J., Zeng, X., Xiao, M., Guo, J., ... & Xu, B. (2021). A deep learning algorithm using CT images to screen for Corona Virus Disease (COVID-19). *European radiology* 1-9.
- [12]. Keidar, D., Yaron, D., Goldstein, E., Shachar, Y., Blass, A., Charbinsky, L., ... & Eldar, Y. C. (2021). COVID-19 classification of X-ray images using deep neural networks. *European radiology* 1-10.
- [13]. Tuncer, T., Ozyurt, F., Dogan, S., & Subasi, A. (2021). A novel Covid-19 and pneumonia classification method based on F-transform. *Chemometrics and Intelligent Laboratory Systems*, 210, 104256.
- [14]. Maniruzzaman, M., Kumar, N., Abedin, M.M., Islam, M.S., Suri, H.S., El-Baz, A.S., & Suri, J.S. (2017). Comparative approaches for classification of diabetes mellitus data: Machine learning paradigm. *Computer methods and programs in biomedicine*, 152, 23-34.
- [15]. Cihan, Ş., Karabulut, B., Arslan, G., & Cihan, G. (2018). Koroner Arter Hastalığı Riskinin Veri Madenciliği Yöntemleri İle İncelenmesi. *Uluslararası Mühendislik Araştırma Ve Geliştirme Dergisi*, 10(1), 85-93.
- [16]. Cihan, Ş., Karabulut, B., Kokoç, M., Arslan, G., Gürel, G. (2019). Analysis of Cryotherapy Treatment of Verruca by Machine Learning. *International Scientific and Vocational Studies Journal*, 3(2), 56-66.
- [17]. Magna, A.A.R., Allende-Cid, H., Taramasco, C., Becerra, C., & Figueroa, R. L. (2020). Application of Machine Learning and Word Embeddings in the Classification of Cancer Diagnosis Using Patient Anamnesis. *IEEE Access* 8, 106198-106213.
- [18]. Nissim, N., Dudaie, M., Barnea, I., Shaked, N.T. (2021). Real-Time Stain-Free Classification of Cancer Cells and Blood Cells Using Interferometric Phase Microscopy and Machine Learning. *Cytometry Part A* 99(5).
- [19]. Arpacı, I., Huang, S., Al-Emran, M., Al-Kabi, M. N., & Peng, M. (2021). Predicting the COVID-19 infection with fourteen clinical features using machine learning classification algorithms. *Multimedia Tools and Applications* 80(8), 11943-11957.
- [20]. Xing, W., & Bei, Y. (2019). Medical health big data classification based on KNN classification algorithm. *IEEE Access* 8, 28808-28819.
- [21]. Ahamad, M. M., Aktar, S., Rashed-Al-Mahfuz, M., Uddin, S., Liò, P., Xu, H., ... & Moni, M. A. (2020). A machine learning model to identify early stage symptoms of SARS-Cov-2 infected patients. *Expert systems with applications* 160, 113661.
- [22]. Hamed, A., Sobhy, A., & Nassar, H. (2021). Accurate Classification of COVID-19 Based on Incomplete Heterogeneous Data using a KNN Variant Algorithm. *Arabian Journal for Science and Engineering* 1-12.

- [23]. Sun, C., Bai, Y., Chen, D., He, L., Zhu, J., Ding, X., ... & Chen, G. (2021). Accurate classification of COVID-19 patients with different severity via machine learning. *Clinical and Translational Medicine* 11(3).
- [24]. Zoabi, Y., Deri-Rozov, S., & Shomron, N. (2021). Machine learning-based prediction of COVID-19 diagnosis based on symptoms. *NPJ Digital Medicine* 4(1), 1-5.
- [25]. Viana Dos Santos Santana, Í., C.M. da Silveira, A., Sobrinho, Á., et al. (2021) Classification Models for COVID-19 Test Prioritization in Brazil: Machine Learning Approach. *Journal of Medical Internet Research*. 2021 Apr;23(4):e27293. DOI: 10.2196/27293. PMID: 33750734; PMCID: PMC8034680.
- [26]. Prasath, VB, Alfeilat, HAA, Lasassmeh, O, Hassanat, A. (2017). Distance and Similarity Measures Effect on the Performance of K-Nearest Neighbors Classifier-A Review. arXiv preprint arXiv:1708.04321.
- [27]. Viswanath, P., & Sarma, T. H. (2011). An improvement to k-nearest neighbor classifier. In 2011 IEEE Recent Advances in Intelligent *Computational Systems* 227-231.
- [28]. Wang, J., Wu, X., Zhang, C. (2005). Support vector machines based on K-means clustering for real-time business intelligence systems. *International Journal of Business Intelligence and Data Mining* 1, 54-64.
- [29]. Kayaalp, N., Arslan, G. (2014). Fuzzy Bayesian Classifier with Learned Mahalanobis Distance. *International Journal of Intelligent Systems* 29, 713-726.
- [30]. Arslan, G., Karabulut, B., Ünver, H.M. (2020). On Using Structural Patterns in Data for Classification, *Advance and Applications in Statistics* 65, 33-56.
- [31]. Karabulut, B., Arslan, G., Ünver, H.M. (2021) Classification Based on Structural Information in Data. *Arabian Journal for Science and Engineering* 1-15.
- [32]. Viana dos Santos Santana, Í.; C. M. da Silveira, A.; Sobrinho, A.; Chaves e Silva, L.; Dias da Silva, L.; Freire de Souza Santos, D.; Candeia, E.; Perkusich, A. (2021), "A Brazilian dataset of symptomatic patients for screening the risk of COVID-19", Mendeley Data, V5, doi: 10.17632/b7zcgmmwx4.5
- [33]. Guha, S., Rastogi, R., & Shim, K. (1998). CURE: an efficient clustering algorithm for large databases. In *ACM Sigmod Record* 27(2), 73-84, ACM.
- [34]. Karypis, G., Han, E. H. S., & Kumar, V. (1999). Chameleon: Hierarchical clustering using dynamic modeling. *Computer* 8, 68-75.
- [35]. Soofi, AA, Awan, A. (2017). Classification Techniques in Machine Learning: Applications and Issues. *Journal of Basic and Applied Sciences* 13, 459-465.
- [36]. Aggarwal, CC. (2014). Instance-Based Learning: A Survey. *Data Classification: Algorithms and Applications* 157.
- [37]. Angiulli, F, Narvaez, E. (2018). Pruning strategies for nearest neighbors competence preservation learners. *Neurocomputing* 308, 8-20.
- [38]. Chicco, D., & Jurman, G. (2023). The Matthews correlation coefficient (MCC) should replace the ROC AUC as the standard metric for assessing binary classification. *BioData Mining*, 16(1), 4.
- [39]. Parvande, S., Yeh, H. W., Paulus, M. P., & McKinney, B. A. (2020). Consensus features nested cross-validation. *Bioinformatics* 36(10), 3093-3098.
- [40]. Varma, S., & Simon, R. (2006). Bias in error estimation when using cross-validation for model selection. *BMC bioinformatics* 7(1), 1-8.
- [41]. Wang, J., Wu, X., Zhang, C. (2005). Support vector machines based on K-means clustering for real-time business intelligence systems. *International Journal of Business Intelligence and Data Mining* 1, 54-64.
- [42]. Lee, S.J., Park, C., Jhun, M., Koo, J.Y. (2007). Support vector machine using K-means clustering. *Journal of the Korean Statistical Society* 36, 175-182.
- [43]. Chen, J., Pan, F. (2010). Clustering-based geometric support vector machines, p. 207-217. In *Proceedings of the Life System Modeling and Intelligent Computing*, Springer, Berlin, Heidelberg
- [44]. Yao, Y., Liu, Y., Yu, Y., et al. (2013). K-SVM: An Effective SVM Algorithm Based on K-means Clustering. *Journal of Computers* 8, 2632-2639.
- [45]. Bang, S., Jhun, M. (2014). Weighted support vector machine using k-means clustering. *Communications in Statistics-Simulation and Computation* 43, 2307-2324
- [46]. Yu, H., Yang, J., Han, J. (2003). Classifying large datasets using SVMs with hierarchical clusters. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. pp. 306-315
- [47]. Horng, S.J., Su, M.Y., Chen, Y.H., et al. (2011). A novel intrusion detection system based on hierarchical clustering and support vector machines. *Expert systems with Applications* 38, 306-313
- [48]. Bang, S., Koo, J.Y., Jhun, M. (2010). Support vector machine using k-spatial medians clustering and recovery process. *Communications in Statistics-Simulation and Computation* 39, 1422-1434